

Reaction rates for reaction-diffusion kinetics on unstructured meshes

Stefan Hellander and Linda Petzold

*Department of Computer Science, University of California,
Santa Barbara, CA 93106-5070 Santa Barbara, USA.*

The reaction-diffusion master equation is a stochastic model often utilized in the study of biochemical reaction networks in living cells. It is applied when the spatial distribution of molecules is important to the dynamics of the system. A viable approach to resolve the complex geometry of cells accurately is to discretize space with an unstructured mesh. Diffusion is modeled as discrete jumps between nodes on the mesh, and the diffusion jump rates can be obtained through a discretization of the diffusion equation on the mesh. Reactions can occur when molecules occupy the same voxel. In this paper, we develop a method for computing accurate reaction rates between molecules occupying the same voxel in an unstructured mesh. For large voxels, these rates are known to be well approximated by the reaction rates derived by Collins and Kimball, but as the mesh is refined, no analytical expression for the rates exists. We reduce the problem of computing accurate reaction rates to a pure preprocessing step, depending only on the mesh and not on the model parameters, and we devise an efficient numerical scheme to estimate them to high accuracy. We show in several numerical examples that as we refine the mesh, the results obtained with the reaction-diffusion master equation approach those of a more fine-grained Smoluchowski particle-tracking model.

I. INTRODUCTION

Spatial stochastic modeling is a tool frequently used to study biochemical reaction networks in cells where the spatial distribution of molecules is non-uniform [1–5]. For instance, reactions can be localized to only a few sites, or molecules may take part in a sequence of reactions where the spatial correlation of newly created molecules affects the dynamics of the whole system [6–8].

The models considered when studying systems on the scale of a living cell are often divided into three levels: the macroscopic level, the mesoscopic level, and the microscopic level. On the macroscopic level the system is modeled by a deterministic partial differential equation (PDE). On the mesoscopic level the system is modeled by the reaction-diffusion master equation (RDME). Exact trajectories can be generated by the next subvolume method (NSM) [9], in which molecules diffuse between voxels at some given intensity, and react at some given intensity when occupying the same voxel. On the microscopic level we track the continuous position and movement of individual molecules, and molecules can react when they are sufficiently close. In this paper we are concerned only with the mesoscopic and microscopic levels.

The RDME is a popular model, evidenced by the number of simulation tools available. While a less detailed model than microscopic models, it has the advantage of being orders of magnitude faster for many biologically relevant problems. Software packages implementing solvers for the RDME include MesoRD [10], PyURDME, StochSS (<http://www.stochss.org>), NeuroRD [11], E-Cell [12] and STEPS [13]. Microscale simulations are suitable when high accuracy is needed, if we need to simulate very diffusion-limited reactions, or if reactions are localized near or on a complex geometrical structure. Software packages implementing microscale solvers include Smoldyn [14], MCell [15], and E-Cell [12].

Many realistic biological systems exist within complex geometries, and a tractable approach to resolving complex geometries is to discretize space using an unstructured mesh. This approach was studied in [16], in which the authors devise a method to obtain accurate diffusive jump rates. The problem of obtaining accurate reaction rates on Cartesian meshes is well studied, see e.g. [7, 8, 17–19], but the problem of obtaining accurate reaction rates for a wide range of voxel sizes for unstructured meshes has not been as thoroughly studied. Isaacson derives a convergent RDME in [20] where the RDME is extended to allow nonlocal reactions and where convergence is to the microscale model proposed by Doi in [21]. Here we consider local reactions only and the Smoluchowski microscale model where molecules are modeled by hard spheres [22] and react according to a Robin boundary condition [23]. Most of our results could be extended to different microscale models.

It is easy to see that for “large enough” voxels on Cartesian meshes we get reaction rates that are simply the effective rate scaled by the volume of the voxel [18], and it is reasonable to assume that the same will be true on unstructured meshes. However, as we show in Sec. IV, if the reactions are diffusion limited this may only be true for quite large voxels, and simulations will be inaccurate as we refine the mesh. This is in contrast to what would be expected numerically; as we refine the mesh we expect simulations to get more accurate, at least up to some maximum spatial resolution [17–19].

One approach to get around this problem is to consider hybrid methods, where mesoscopic simulations can be performed on fairly coarse meshes for most species, combined with accurate microscopic simulations for some species whose dynamics need to be simulated at a high spatial resolution [24–27].

In this paper we instead develop a method to efficiently compute accurate reaction rates for the RDME on unstructured meshes for a wide range of voxel sizes and for diffusion-limited reactions, thus facilitating accurate simulations of biochemical systems in complex geometries. We show that the common approach of simply scaling the bimolecular reaction rates by the volume of the voxels leads to inaccurate results for some systems, and that by instead following the approach outlined in this paper, the RDME, for many problems, converges to the corresponding microscale results.

The remainder of the paper is organized as follows. In Sec. II we introduce the mesoscopic and microscopic modeling frameworks. In Sec. III we describe a method to compute mesoscopic reaction rates, and in Sec. IV we show with numerical examples first how to choose the parameters of the method and then that the method itself yields accurate reaction rates leading to convergence of the RDME.

II. BACKGROUND

In this section we introduce the mesoscopic RDME and the microscopic Smoluchowski model. Throughout this paper, we consider the accuracy of the RDME relative to the Smoluchowski model, but note that most of the derivations in Sec. III would be valid also for a different microscale model; we select a specific model for convenience.

As we will see in Sec. III, to derive accurate reaction rates for the RDME it is useful to consider individual trajectories of the stochastic system instead of the full probability distribution. In addition to introducing the full models, we therefore also introduce methods for generating exact trajectories of a system.

A. Mesoscopic simulations

Consider a volume Ω divided into N non-overlapping voxels, and a reaction network consisting of S species. We denote the $N \times S$ state matrix by \mathbf{x} , where the i -th row, \mathbf{x}_i , gives the species copy numbers of voxel i , while the j -th column, \mathbf{x}_j , gives the copy numbers of species j for each voxel. Assume that the system has M reactions. We denote the propensity function for reaction r in voxel i by $a_{ir}(\mathbf{x}_i)$, and the stoichiometry vector for reaction r in voxel i by $\boldsymbol{\mu}_{ir}$. We denote the propensity function for a diffusive jump by species j from voxel i to k by $d_{ijk}(\mathbf{x}_i)$, and the associated stoichiometry vector by $\boldsymbol{\nu}_{ijk}$.

The RDME describes the time evolution of p :

$$\begin{aligned} \frac{d}{dt}p(\mathbf{x}, t) = & \sum_{i=1}^N \sum_{r=1}^M a_{ir}(\mathbf{x}_i - \boldsymbol{\mu}_{ir})p(\mathbf{x}_1, \dots, \mathbf{x}_i - \boldsymbol{\mu}_{ir}, \dots, \mathbf{x}_N, t) - \sum_{i=1}^N \sum_{r=1}^M a_{ir}(\mathbf{x}_i)p(\mathbf{x}, t) \\ & + \sum_{j=1}^M \sum_{i=1}^N \sum_{k=1}^N d_{jik}(\mathbf{x}_j - \boldsymbol{\nu}_{ijk})p(\mathbf{x}_1, \dots, \mathbf{x}_j - \boldsymbol{\nu}_{ijk}, \dots, \mathbf{x}_N, t) \\ & - \sum_{j=1}^M \sum_{i=1}^N \sum_{k=1}^N d_{ijk}(\mathbf{x}_j)p(\mathbf{x}, t). \end{aligned} \quad (1)$$

The RDME is in general too high-dimensional to be solved by direct methods; a common approach is instead to generate statistics of the system using a Monte Carlo scheme. Exact trajectories of the RDME can be generated as follows. A molecule with diffusion rate D can jump to adjacent voxels at intensity γ_D , where $\gamma_D = 2dD/h^2$ on a Cartesian mesh in d dimensions, and where γ_D can be determined from e.g. a finite element discretization in the case of unstructured meshes [16]. Molecules can undergo unimolecular and bimolecular reactions; a pair of molecules are only allowed to react when occupying the same voxel.

The next event of a system can be determined by sampling the tentative time of every possible diffusion and reaction event; every tentative event time is assumed to be exponentially distributed, and the smallest tentative event time determines which event will fire next.

An efficient algorithm to generate exact trajectories is the next subvolume method (NSM) [9].

B. Microscopic simulations

On the microscopic scale two molecules A and B , modeled as hard spheres with radii σ_A and σ_B , diffuse with diffusion constants D_A and D_B and react according to the Smoluchowski equation with a Robin boundary condition at the reaction radii $\sigma = \sigma_A + \sigma_B$. Let $\mathbf{r} = \mathbf{x}_1 - \mathbf{x}_2$ be the relative position of the two molecules (molecule A has position \mathbf{x}_1 and molecule B has position \mathbf{x}_2). Then the equation governing the relative position of the molecules is

$$\frac{\partial p}{\partial t} = D\Delta p(\mathbf{x}, t), \quad (2)$$

with a Robin boundary condition [23, 28] given by

$$K \frac{\partial p}{\partial n} \Big|_{|\mathbf{x}|=\sigma} = k_a p(|\mathbf{x}| = \sigma, t), \quad (3)$$

where

$$K = \begin{cases} 4\pi\sigma^2 D, & (3D) \\ 2\pi\sigma D, & (2D). \end{cases} \quad (4)$$

It can be shown that $\mathbf{R} = \mathbf{x}_1 + \mathbf{x}_2$ moves according to normal diffusion [29].

A system of more than two molecules becomes an intractable many-body problem. A popular method to simulate such systems is the GFRD algorithm [29, 30]. Instead of considering the full many-body problem, the system is divided into subsets of one-body and two-body problems by selecting a time step Δt during which each such subset is unlikely to interact with any other subset of molecules. Now, during Δt , molecules are either propagated by normal

diffusion, or in the case of pairs of molecules, by sampling a new \mathbf{r} and \mathbf{R} . All microscale results in Sec. IV were obtained with the algorithm developed in [31].

Another approach to simulating Smoluchowski dynamics is to select a fixed time step during which individual molecules are propagated by normal diffusion. Given information about each molecule's initial and final positions, it is possible to compute the probabilities of molecules reacting [14, 15, 32].

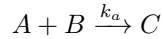
III. REACTION RATES

It has been shown in the case of structured Cartesian meshes that it is important to choose the reaction rates carefully to obtain accurate simulations [8, 17, 18]. For Cartesian meshes, the reaction rates derived in [17–19] for the standard RDME, and in [8] for a generalized RDME, were shown to yield accurate results down to a lower bound on the mesh resolution on the order of a few times the reaction radius for the standard RDME, and down to a mesh resolution on the order of the reaction radius in the case of the generalized RDME (in which molecules occupying neighboring voxels can react).

While Cartesian meshes are good for simple simulation volumes, they are less suitable for complex geometries. In that case, unstructured triangular (2D) or tetrahedral (3D) meshes are preferable. However, the analytical expressions for the reaction rates derived in [8, 17–19] depend on the assumption of a Cartesian mesh.

In this section we devise an efficient way to numerically compute accurate reaction rates for the RDME on unstructured meshes.

We consider a system of three species A , B , and C undergoing the single irreversible reaction



in a domain Ω , where k_a is the microscopic reaction rate. The A molecule does not diffuse, while the B molecule diffuses with diffusion rate D . We denote the reaction radius of the A and the B molecule by σ . Given these microscopic parameters, we seek to obtain the corresponding mesoscopic reaction rate k_a^{meso} , where k_a^{meso} is the rate, in units of s^{-1} , at which molecules react when they occupy the same voxel.

To this end, we make the assumption that the mean binding time on the microscale, $\tau_{\text{micro}}(k_a)$, and the mesoscale, $\tau_{\text{meso}}(k_a^{\text{meso}})$, should match. For a given k_a we want to find k_a^{meso} such that

$$\tau_{\text{micro}}(k_a) = \tau_{\text{meso}}(k_a^{\text{meso}}) \quad (5)$$

holds.

A. Cartesian meshes

We first briefly summarize the derivation of reaction rates on Cartesian meshes, as some of the methodology carries over to the case of unstructured meshes. Throughout this section the domain will be a cube of volume V , discretized into N voxels, and where h denotes the width of a voxel.

1. Microscopic effective binding time

The effective binding time, τ_{micro} , can be divided into two parts: an initial diffusion part and a reaction part. We assume that the B molecule has a uniform initial distribution and that the fixed A molecule is some distance away from the boundary, and we denote the time it takes until the molecules are in contact for the first time by $\tau_{\text{diff}}^{\text{micro}}$. The reaction part is defined to be the time that remains until they react; that is, the time until the molecules react given that they start in contact. We denote the reaction part by $\tau_{\text{react}}^{\text{micro}}$. By definition we have

$$\tau_{\text{micro}} = \tau_{\text{diff}}^{\text{micro}} + \tau_{\text{react}}^{\text{micro}}. \quad (6)$$

Let σ be the reaction radius, and D the diffusion constant of the B molecule. We know that [7, 18]

$$\tau_{\text{diff}}^{\text{micro}} \approx \begin{cases} \frac{V}{4\pi\sigma D}, & (3D) \\ \frac{V \left\{ \log \left(\pi^{-1} \frac{V^{1/2}}{\sigma} \right) \right\}}{2\pi D}, & (2D) \end{cases} \quad (7)$$

and that

$$\tau_{\text{react}}^{\text{micro}} = \frac{V}{k_a} \quad (2D, 3D). \quad (8)$$

Let $k_{\text{CK}} = 4\pi\sigma D k_a / (4\pi\sigma D + k_a)$. In 3D we obtain the well-known expression [33, 34] for the mean binding time, given by

$$\tau_{\text{micro}} = \frac{V}{4\pi\sigma D} + \frac{V}{k_a} = \frac{4\pi\sigma D + k_a}{4\pi\sigma D k_a} V =: \frac{V}{k_{\text{CK}}}, \quad (9)$$

where k_{CK} is the Collins and Kimball rate [33]. A common approach in mesoscopic simulations is to let the mesoscopic rate k_a^{meso} be the Collins and Kimball rate scaled by the volume of the voxel, V_{vox} , that is $k_a^{\text{meso}} = k_{\text{CK}}/V_{\text{vox}}$.

2. Mesoscopic effective binding time

The mean binding time on the mesoscopic scale can be divided into two parts in a way analogous to the microscale case. The diffusion part, $\tau_{\text{diff}}^{\text{meso}}$, is the average time until the B molecule reaches the voxel that is occupied by the A molecule. The reaction part, $\tau_{\text{react}}^{\text{meso}}$, is the time until the A and the B molecule react, given that they start in the same voxel. Thus

$$\tau_{\text{meso}} = \tau_{\text{diff}}^{\text{meso}} + \tau_{\text{react}}^{\text{meso}}. \quad (10)$$

These quantities are both known analytically on a Cartesian mesh [17, 18]:

$$\tau_{\text{diff}}^{\text{meso}} = \begin{cases} \frac{C_3 V}{6Dh} + O\left(N^{\frac{1}{2}}\right) & (3D) \\ \frac{V}{4\pi D} \log(N) + \frac{C_2 V}{4D} + O\left(N^{-1}\right) & (2D) \end{cases} \quad (11)$$

and

$$\tau_{\text{react}}^{\text{meso}} = \frac{N}{k_a^{\text{meso}}}. \quad (12)$$

Now let

$$C_d \approx \begin{cases} 0.1951, & d = 2 \\ 1.5164, & d = 3. \end{cases} \quad (13)$$

By inserting (7), (8), (11), (12) into (5), and solving for k_a^{meso} , we obtain expressions for the mesoscopic reaction rate k_a^{meso} :

$$k_a^{\text{meso}} = \frac{k_a}{h^d} \left(1 + \frac{k_a}{D} G(h, \sigma) \right)^{-1} \quad (14)$$

where

$$G(h, \sigma) = \begin{cases} \frac{1}{2\pi} \log\left(\pi^{-\frac{1}{2}} \frac{h}{\sigma}\right) - \frac{1}{4} \left(\frac{3}{2\pi} + C_2\right) & (2D) \\ \frac{1}{4\pi\sigma} - \frac{C_3}{6h} & (3D). \end{cases} \quad (15)$$

In [18] we showed that, in general, the most accurate simulations in 2D and 3D are obtained for

$$h_{\infty}^* = \begin{cases} \frac{2C_3}{3D} \pi \sigma \approx 3.2\sigma, & (3D) \\ \sqrt{\pi} e^{\frac{3+2C_2\pi}{4}} \sigma \approx 5.1\sigma, & (2D) \end{cases} \quad (16)$$

where the RDME may get less accurate as the mesh is refined further. The lower limit h_{∞}^* is also the smallest mesh size for which $k_a^{\text{meso}} > 0$ as $k_a \rightarrow \infty$.

Note that by (10) and (12), the mesoscopic rate can be written as

$$k_a^{\text{meso}} = \frac{N}{\tau_{\text{micro}} - \tau_{\text{diff}}^{\text{meso}}}. \quad (17)$$

If $\tau_{\text{micro}} \gg \tau_{\text{diff}}^{\text{meso}}$, then $k_a^{\text{meso}} \approx k_{\text{CK}}/h^3$ in 3D, and this is an assumption sometimes made in software implementations of the RDME. As we will see in Sec. IV, this assumption is not always satisfied and when not, this rate yields inaccurate results.

B. Unstructured meshes

In the case of Cartesian meshes we have analytical expressions for the reaction rates, where the derivation is based on analytical expressions for $\tau_{\text{diff}}^{\text{meso}}$ and $\tau_{\text{react}}^{\text{meso}}$. For unstructured meshes we can consider the same setup as for Cartesian meshes, but we do not have analytical expressions for $\tau_{\text{diff}}^{\text{meso}}$ and $\tau_{\text{react}}^{\text{meso}}$, and instead we need to compute them numerically.

The quantity $\tau_{\text{diff}}^{\text{meso}}$ can be computed independently of the microscopic reaction rates, and depends only on the diffusion rate and the mesh.

Now assume that the A molecule has diffused such that it occupies the same voxel as the B molecule. Denote by d_{tot} the total diffusion rate out of the voxel, as obtained by e.g. a finite element discretization of the diffusion equation. The molecules then react with probability

$$p_r = \frac{k_a^{\text{meso}}}{k_a^{\text{meso}} + d_{\text{tot}}}, \quad (18)$$

and diffuse apart with probability $1 - p_r$. The molecules, on average, occupy the same voxel p_r^{-1} times before they react.

Given that the molecules occupy the same voxel, the average time until the next event will be given by

$$\frac{1}{k_a^{\text{meso}} + d_{\text{tot}}}. \quad (19)$$

They react with probability p_r . If they do not react, they consequently diffuse apart, and will then occupy adjacent voxels. Denote by t_1 the average time until the molecules again occupy the same voxel. We can summarize the above process as follows:

Assume that the molecules occupy the same voxel.

- (1) With probability p_r they react after an average time of $1/(k_a^{\text{meso}} + d_{\text{tot}})$.
- (2) With probability $1 - p_r$ they do not react. They occupy the same voxel once again after an average time of $1/(k_a^{\text{meso}} + d_{\text{tot}}) + t_1$.

The molecules react after occupying the same voxel on average p_r^{-1} times.

We obtain

$$\tau_{\text{react}}^{\text{meso}} = \frac{1}{p_r} \left[p_r \frac{1}{k_a^{\text{meso}} + d_{\text{tot}}} + (1 - p_r) \left(\frac{1}{k_a^{\text{meso}} + d_{\text{tot}}} + t_1 \right) \right], \quad (20)$$

which, after some straightforward algebra, yields

$$\tau_{\text{react}}^{\text{meso}} = \frac{1}{k_a^{\text{meso}}} (1 + d_{\text{tot}} t_1). \quad (21)$$

Now, to satisfy $\tau_{\text{meso}} = \tau_{\text{micro}}$, we should find k_a^{meso} such that

$$\tau_{\text{micro}} = \tau_{\text{diff}}^{\text{meso}} + \tau_{\text{react}}^{\text{meso}}, \quad (22)$$

which holds if and only if

$$k_a^{\text{meso}} = \frac{1 + d_{\text{tot}} t_1}{\tau_{\text{micro}} - \tau_{\text{diff}}^{\text{meso}}}. \quad (23)$$

Thus, to obtain the reaction rate k_a^{meso} for the molecules in a voxel V_i , we must compute $\tau_{\text{diff}}^{\text{meso}}$ and t_1 .

C. Computing $\tau_{\text{diff}}^{\text{meso}}$ and t_1

Assume that the A molecule occupies a voxel V_i . We first compute $\tau_{\text{diff}}^{\text{meso}}$. A straightforward approach would be a simple Monte Carlo procedure:

Algorithm 1

1. Initialize the B molecule according to a uniform distribution on the mesh.
 2. Simulate the system until the B molecule finds V_i .
 3. Repeat N_1 times and compute the mean.
-

However, for fine mesh resolutions the naive approach becomes computationally expensive; on a Cartesian mesh we know that the average number of steps required to find V_i scales proportionally to the number of voxels.

Instead we propose the following algorithm: In step 2, instead of simulating the B molecule until it finds V_i , we note that the molecules will be effectively uniformly distributed after some time Δt_1 , where $\sqrt{2dD\Delta t_1} \sim |\Omega|^{\frac{1}{d}}$. Thus, if the B molecule has not reached V_i after $\Delta t_1 \sim \frac{|\Omega|^{\frac{2}{d}}}{2dD}$, it is approximately uniformly distributed, and the mean time remaining until the B molecule finds V_i is given by $\tau_{\text{diff}}^{\text{meso}}$. Let $\tau_{t \leq \Delta t}^{\text{reac}}$ denote the average time required to find V_i , given that the B molecule reaches V_i before time Δt_1 . Let q denote the probability that the B molecule finds V_i before Δt_1 . Then

$$\tau_{\text{diff}}^{\text{meso}} \approx q\tau_{t \leq \Delta t}^{\text{reac}} + (1 - q)(\Delta t_1 + \tau_{\text{diff}}^{\text{meso}}). \quad (24)$$

By solving for $\tau_{\text{diff}}^{\text{meso}}$ we obtain

$$\tau_{\text{diff}}^{\text{meso}} \approx \tau_{t \leq \Delta t}^{\text{reac}} + \frac{1 - q}{q} \Delta t_1. \quad (25)$$

Thus, by letting $\Delta t_1 = c_1 \frac{|\Omega|^{\frac{2}{d}}}{2dD}$ for some suitable constant c_1 , we obtain an estimate of $\tau_{\text{diff}}^{\text{meso}}$ by computing $\tau_{t \leq \Delta t}^{\text{reac}}$ and q . We discuss how to choose c_1 in Sec. IV.

We now wish to estimate t_1 . Again we could approach this with a naive Monte Carlo approach:

Algorithm 2

1. Sample the initial position of the B molecule by letting it diffuse from V_i to an adjacent voxel.
 2. Diffuse the B molecule until it finds V_i and record the time t .
 3. Repeat (1)-(2) N_2 times and compute the mean.
-

However, again we note that if the B molecule does not reach V_i after $\Delta t_2 = c_2 \frac{|\Omega|^{\frac{2}{d}}}{2dD}$, for some suitably chosen constant c_2 , it will be approximately uniformly distributed on Ω . Thus, after a time Δt_2 the time remaining can be approximated by $\tau_{\text{diff}}^{\text{meso}}$. Now, in step 2 above, we see that it is enough to simulate a trajectory until time Δt_2 ; if the B molecule has not visited V_i , we simply add $\tau_{\text{diff}}^{\text{meso}}$ to the total time.

We summarize the algorithm for computing $\tau_{\text{diff}}^{\text{meso}}$ and t_1 in Alg. 3.

Algorithm 3

Assume that the A molecule occupies voxel V_i .

1. Initialize the B molecule uniformly on Ω .
2. Let the B molecule diffuse until
 - (a) It finds V_i , or
 - (b) $t = c_1 \frac{|\Omega|^{\frac{2}{d}}}{2dD}$.
3. Repeat (1)-(2) N_1 times.
4. Let q denote the proportion of trajectories that ended in 2 (a), and let $\tau_{t \leq \Delta t}^{\text{reac}}$ denote the average time of the trajectories that ended in 2 (a). Estimate $\tau_{\text{diff}}^{\text{meso}}$ by

$$\widehat{\tau_{\text{diff}}^{\text{meso}}} = \tau_{t \leq \Delta t}^{\text{reac}} + \frac{1-q}{q} \Delta t_1. \quad (26)$$

5. Initialize the B molecule in a voxel adjacent to V_i , proportionally to the diffusion rates out of V_i .
6. Let the B molecule diffuse until
 - (a) It reaches V_i after some time $t_{\text{reac}} \leq \Delta t_2$. Let $\tilde{t}_1 = t_{\text{reac}}$ be an approximation of t_1 .
 - (b) $t = c_2 \frac{|\Omega|^{\frac{2}{d}}}{2dD}$. Let $\tilde{t}_1 = \Delta t_2 + \widehat{\tau_{\text{diff}}^{\text{meso}}}(k_a^{\text{meso}})$ be an approximation of t_1 .
7. Repeat (5)-(6) N_2 times.
8. Estimate t_1 with the mean of \tilde{t}_1 .
9. Estimate k_a^{meso} by

$$\tilde{k}_a^{\text{meso}} = \frac{1 + d_{\text{tot}} \tilde{t}_1}{\tau_{\text{micro}} - \widehat{\tau_{\text{diff}}^{\text{meso}}}}. \quad (27)$$

Note that for a given mesh we only need to compute $\tau_{\text{diff}}^{\text{meso}}$ and t_1 once, even if different species have different diffusion rates. This is because $\tau_{\text{diff}}^{\text{meso}}$ and t_1 are inversely proportional to the diffusion constant.

D. Linear approximation of $1/k_a^{\text{meso}}$

In principle we should compute k_a^{meso} for each voxel of the mesh, but for a fine mesh this will be prohibitively expensive computationally, as the number of voxels can be on the order of 10^5 or more. However, while k_a^{meso} has a nontrivial dependence on the volume of the voxels, we can still assume that locally k_a^{meso} is approximately inversely proportional to the volume. That is, by assuming that the distribution of volumes in a mesh is not too wide, we can estimate the mesoscopic reaction rate by

$$(k_a^{\text{meso}})^{-1} = k_0 + k_1 V_{\text{vox}}, \quad (28)$$

where V_{vox} is the volume of a voxel. For (28) to be valid, we also have to assume that for a given mesh, k_a^{meso} is approximately the same for two different voxels of the same volume. In practice this puts some regularity constraints on the domain. This condition could be violated for very complex domains, in which case we could compute the coefficient of determination to evaluate how good of an approximation the linear regression provides. In Sec. IV we show that the assumption (28) is a good approximation for a few different common geometries.

For very small voxels in a mesh we may have

$$V_{\text{vox}} < -\frac{k_0}{k_1}, \quad (29)$$

leading to a negative estimate of k_a^{meso} . If this happens for many voxels in a mesh, then the mesh is over resolved, analogously to how a Cartesian mesh can be over resolved for $h < h_\infty^*$. It may however happen for a few voxels in a mesh, since the voxel volume is non-uniform, without the mesh overall being over resolved. In this case we somewhat arbitrarily compute the rate for that voxel with $V_{\text{vox}} = -\frac{k_0}{k_1} + \epsilon$ for some small positive ϵ , to force the rate to be positive. This will introduce a small error, but as we will see in the second example in Sec. IV, it can be neglected even for very fine meshes.

IV. NUMERICAL RESULTS

A. Accuracy of the method

We need to determine suitable values for the constants c_1 and c_2 . To that end we consider the simplified setup of one A molecule fixed to a single voxel near the center of a sphere of radius 1. One B molecule reacts irreversibly with the A molecule at the reaction radius $\sigma = 5 \cdot 10^{-3}$ and diffuses with diffusion constant $D = 1$. The microscopic reaction rate is 1.0.

We first compute highly accurate approximations, \bar{t}_1 and $\overline{\tau_{\text{diff}}^{\text{meso}}}$, of t_1 and $\tau_{\text{diff}}^{\text{meso}}$ using Algs. 1 and 2. We then proceed to compute approximations of t_1 and $\tau_{\text{diff}}^{\text{meso}}$, \tilde{t}_1 and $\widetilde{\tau_{\text{diff}}^{\text{meso}}}$, using Alg. 3 while varying c_1 and c_2 . The relative errors $E_1 = |\tilde{t}_1 - \bar{t}_1|/|\bar{t}_1|$ and $E_2 = |\widetilde{\tau_{\text{diff}}^{\text{meso}}} - \overline{\tau_{\text{diff}}^{\text{meso}}}|/|\overline{\tau_{\text{diff}}^{\text{meso}}}|$ are plotted as heat maps in Fig. 1. In Fig. 2 we show that the approximate method (Alg. 3) gives a speed-up of up to an order of magnitude compared to the exact approach of Algs. 1 and 2.

We repeated the computations for a sequence of meshes of different resolutions (utilizing the tool Gnu Parallel [35]), and as we can see, $c_1 = 5$ and $c_2 = 5$ give errors on the order of 1 – 2%, thus being reasonable choices.

It is reasonable to assume that Eqs. (13)-(15), with h substituted for $V_{\text{vox}}^{1/3}$, where V_{vox} is the volume of a voxel, will agree quite well for spatial reactions in simpler geometries. To test this hypothesis, as well as to show that the linearity assumption in Sec. IV is reasonable, we computed the rates numerically according to Alg. 3, and compared the results with the rates computed according to (13)-(15). In Fig. 3 we see that the rates do indeed agree quite well, while the rates computed as $k_{\text{CK}}/V_{\text{vox}}$ become increasingly incorrect as the mesh is refined.

Note that we cannot expect the numerical approach to always agree with Eqs. (13)-(15). The reason is that the formula (13)-(15) does not take into account that voxels in an unstructured mesh may be of different sizes; the rate for a small voxel within a mesh of mostly larger voxels may be incorrectly approximated by (13)-(15).

We should also note that the microscopic mean binding time for voxels close to a reflective boundary will not be given by τ_{micro} as computed by Eqs. (6)-(8) (the mean binding time for a voxel far from the boundary). Thus, if very high accuracy is needed, we should for those voxels compute also τ_{micro} numerically. However, since we take a sample of voxels and perform linear regression, the effect of that error will generally be small, and as we show in the next example, we obtain very accurate simulations also when neglecting this error.

B. Dissociation with fast rebinding

To demonstrate the applicability of the method we consider a system where the microscale model displays dynamics different from the mesoscopic model. A simple example is given by



On the microscopic scale, following a dissociation of S_1 , the products S_{11} and S_{12} are placed in contact. Thus, the probability of S_{11} and S_{12} to rebind quickly and form the complex S_2 is higher than on the mesoscopic scale, where S_{11} and S_{12} are assumed to be well-mixed inside a voxel immediately following a dissociation.

With the RDME we expect to approach the microscale results as we refine the mesh, and ideally, for the finest mesh sizes, we hope to reproduce the results of the microscale simulations to high accuracy.

We simulate the system defined by Eq. (30) for 2s, sampling the state at $M = 200$ uniformly distributed time

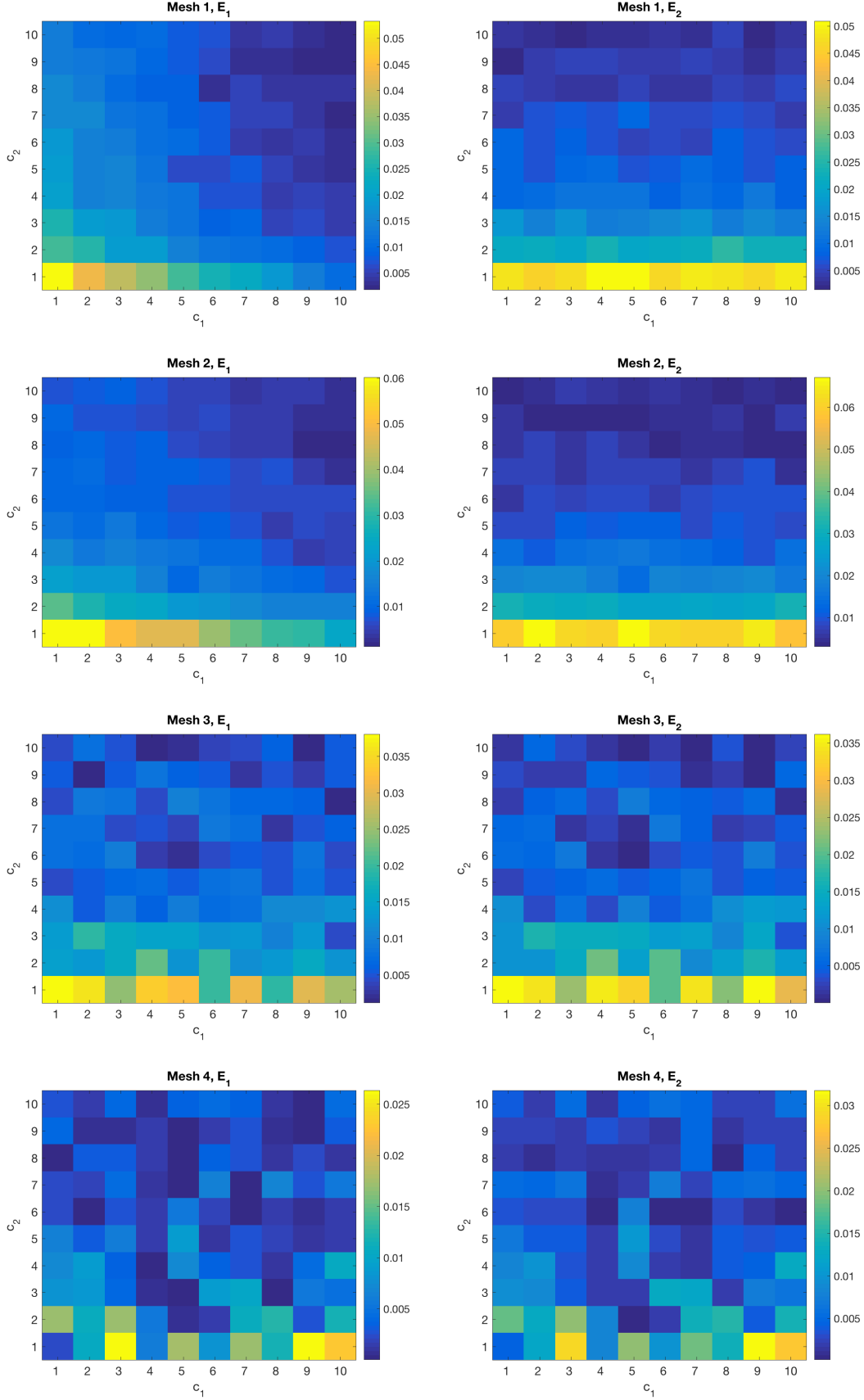


FIG. 1. The errors E_1 and E_2 as functions of c_1 and c_2 , for four different meshes of increasing resolution. From top to bottom: mesh 1: 358 voxels, mesh 2: 1185 voxels, mesh 3: 22644 voxels, and mesh 4: 98206 voxels. The error is computed as the mean of 10^6 trajectories. The stochastic error is fairly small, and as we can see, E_1 and E_2 are generally on the order of 1%-2% for $c_1 = c_2 = 5$, which we therefore argue is a reasonable choice.

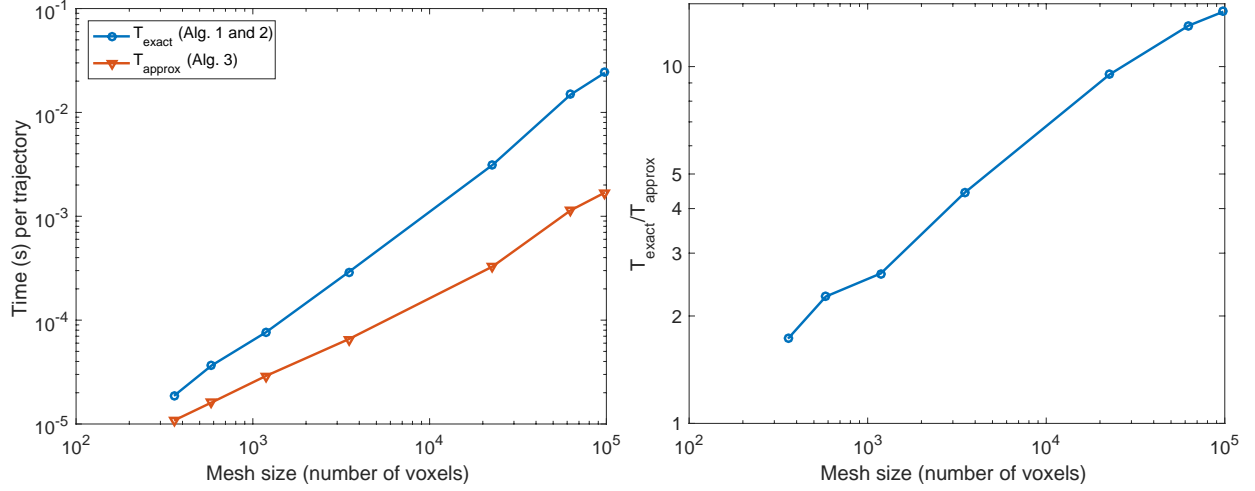


FIG. 2. To the left we plot the time in seconds per trajectory for the exact method, T_{exact} , outlined in Algs. 1 and 2 (blue line with circles) and the time per trajectory of the approximate method, T_{approx} , outlined in Alg. 3 (red line with triangles) with $c_1 = c_2 = 5$. To the right we plot the relative speed-up with $c_1 = c_2 = 5$. As we can see, for coarse meshes the speed-up is fairly modest (about two times), but as we refine the mesh the speed-up becomes significant. For a fine mesh consisting of 10^5 voxels, we obtain a speed-up of almost 15.

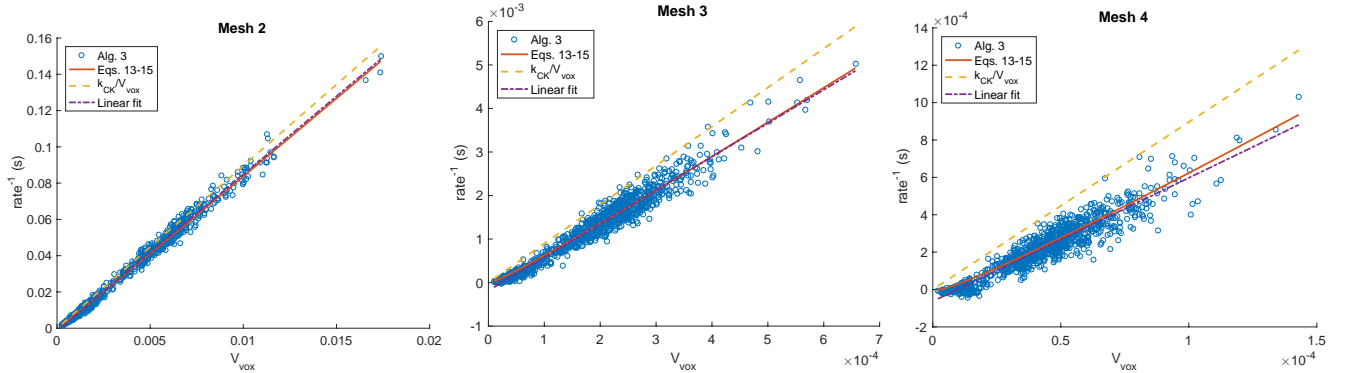


FIG. 3. The inverse of the rates sampled according to Alg. 3 (blue circles), the linear fit of these samples (dashed-dotted purple line), rates computed according to Eqs. (13)-(15) with $h = V_{\text{vox}}^{1/3}$ (solid red line), and the effective rate scaled by the volume of the voxels (dashed yellow line). We computed 10^3 samples, with 10^3 trajectories for each sample. The total execution time on a desktop computer with an Intel i7-4770 CPU at 3.50GHz running Ubuntu 14.04 was 8 s for the coarse mesh (mesh 2, $\sim 10^3$ voxels), 83 s for the intermediate mesh (mesh 3, $\sim 23 \cdot 10^3$ voxels), and 339 s for the fine mesh (mesh 4, $\sim 10^5$ voxels). Again, $c_1 = c_2 = 5$. As we can see, the numerically computed rates agree reasonably well with the rates obtained with Eqs. (13)-(15) for all three mesh sizes, but the difference does increase as the mesh is refined. For the coarser mesh we see that $k_{\text{CK}}/V_{\text{vox}}$ provides a decent approximation, but as we refine the mesh it becomes a poor approximation of the reaction rates, in agreement with the theory outlined in Sec. III A 2.

points with $t_1 = 0.01$ and $t_{200} = 2.0$, and compute the average relative l_1 error of the final product S_2 as

$$E = \frac{1}{M} \sum_{i=1}^M \frac{|z_i - y_i|}{|y_i|}, \quad (31)$$

where z_i is the average of 5000 mesoscopic trajectories at time t_i , and y_i is the average of 10000 trajectories on the microscopic scale at time t_i . The initial number of S_1 molecules is 100, and the initial number of S_{11} , S_{12} , and S_2 molecules is 0.

The microscale parameters are given by

$$\begin{cases} \sigma_1 = \sigma_{11} = \sigma_{12} = \sigma_2 = 2.5 \cdot 10^{-3} \\ D_1 = D_{11} = D_{12} = D_2 = 1.0 \\ k_1 = 10.0 \\ k_2 = 1.0, \end{cases} \quad (32)$$

and the volume of each domain is $V = 1$ (see Fig. 4).

To demonstrate the flexibility of the algorithm, we consider three different geometries: (a) a cube, (b) a sphere, and (c) two half-spheres connected by a cylinder. We start out with a coarse mesh, and then consider successively finer meshes. In Fig. 5 we show that for the finest meshes the relative error is on the order of a few percent, in contrast to the constant high error when the rate is given by k_{CK}/V_{vox} . We can also see that the convergence is similar for all three geometries. When computing the rates we neglected the error introduced by approximating the mean binding time close to boundaries by τ_{micro} , and as we can see, it did not introduce a large error in the simulated dynamics of the system.

It is noteworthy that, for this particular system, the microscale simulations will be fairly efficient in comparison to the mesoscopic simulations for the finest mesh sizes. For instance, for the finest cubic mesh ($\sim 240,000$ voxels), a single trajectory on the mesoscopic scale (excluding the preprocessing time) is more expensive to simulate than the corresponding microscale simulation of a trajectory (7.35s vs 5.83s on a desktop computer with an Intel i7-4770 CPU at 3.50GHz running Ubuntu 14.04). However, this system has a fairly low number of molecules, making it well suited for microscale simulations. Also, the mesoscale simulations scale quadratically with the number of voxels. Thus, for systems with more molecules or for less resolved geometries, the RDME will be competitive.

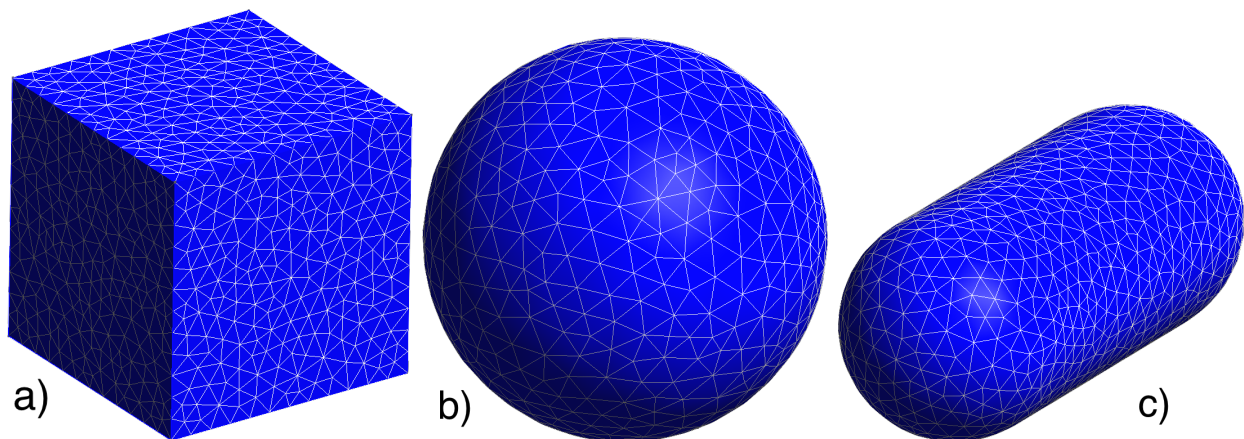


FIG. 4. We consider three different geometries: (a) a cube, (b) a sphere, and (c) two half-spheres connected by a cylinder. Each has a total volume of 1, and in geometry (c) the ratio of the radius to the length of the cylinder is 3. The geometries are discretized into tetrahedral meshes using Gmsh [36].

V. DISCUSSION

It has been shown that it is crucial for the accuracy of the RDME that reaction rates are selected with care when reactions are diffusion limited or when the spatial resolution is high. For structured Cartesian meshes, this problem has been studied in some detail, but not for simulations on unstructured meshes.

We have devised a method to compute accurate rates on unstructured meshes, and shown in numerical examples that with these rates the RDME is accurate also on unstructured meshes, for a wide range of mesh sizes and reaction rates. We have also shown for a few different geometries that the numerically computed rates agree well with the corresponding rates on Cartesian meshes, suggesting that for many systems the analytically derived rates on Cartesian meshes will provide sufficient accuracy.

For complex geometries, however, such an assumption cannot be made in general. It is also worth noting that the numerical approach outlined in this paper can be generalized to other types of reactions; we may for instance consider reactions between molecules in 3D and complex surfaces. It is straightforward to extend the algorithm to

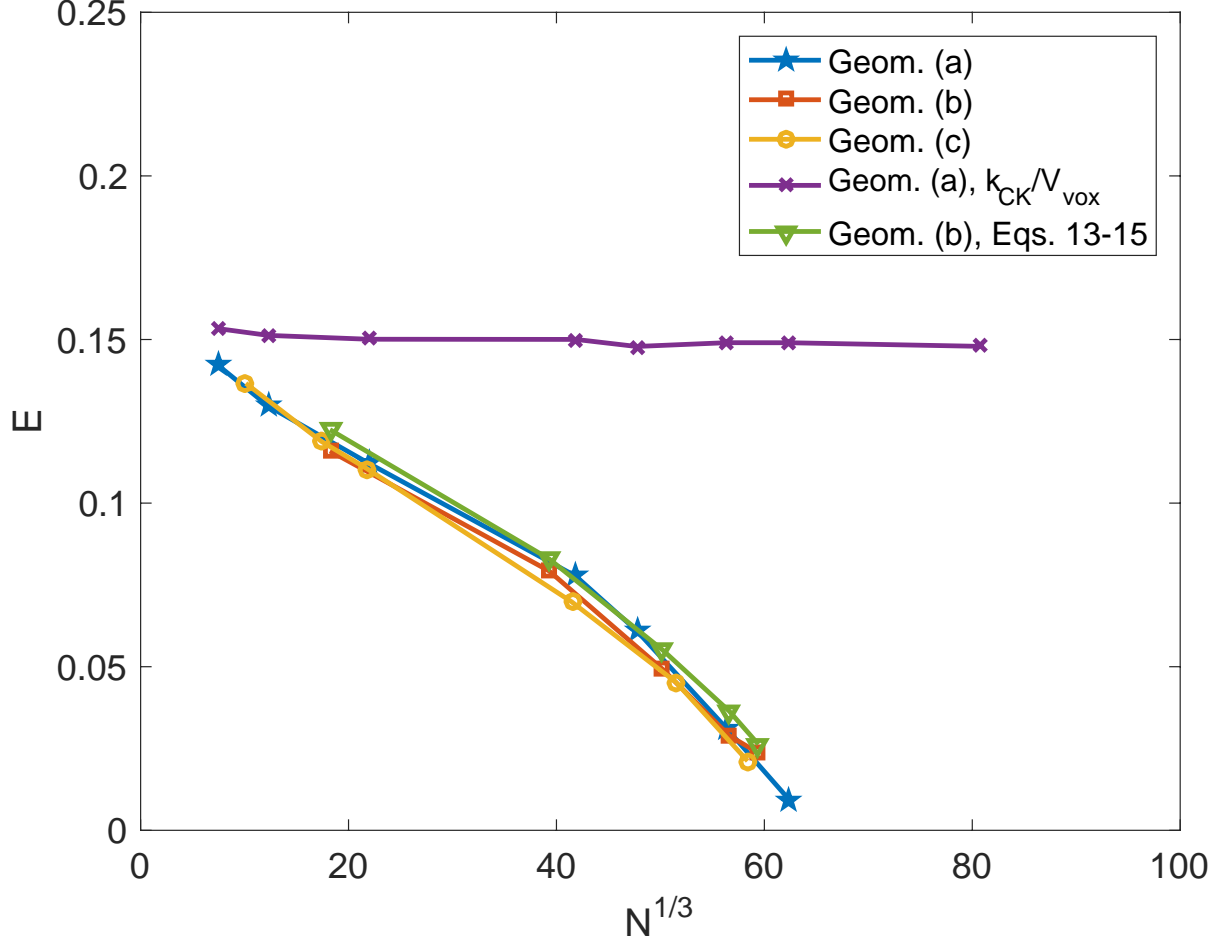


FIG. 5. The relative error E as defined by (31), as a function of the total number of voxels N . With the reaction rate k_2^{meso} computed according to Alg. 3, the error decreases as we increase the resolution of the mesh (geometry (a): blue line with stars, geometry (b): red line with squares, and geometry (c) yellow line with circles), while we see no convergence if we choose the reaction rate to be the effective rate scaled by the volume of the voxel (purple line with crosses). We also see that for this problem we obtain accurate simulations also with the rates computed according to Eqs. (13)-(15) with $h = V_{\text{vox}}^{1/3}$ (green line with triangles).

such reactions, where analytical approaches are less likely to be successful.

VI. ACKNOWLEDGMENTS

This work was supported by NIBIB of the NIH under Grant No. R01-EB014877-01, the Institute for Collaborative Biotechnologies through Grant No. W911NF-09-D-0001 from the U.S. Army Research Office, and U.S. DOE Grant No. de-sc0008975. The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

-
- [1] M. J. Lawson, B. Drawert, M. Khammash, and L. Petzold. Spatial stochastic dynamics enable robust cell polarization. *PLoS Comput. Biol.*, 9(7):e1003139, 2013.
 - [2] A. D. Rutenberg M. Howard. Pattern formation inside bacteria: fluctuations due to the low copy number of proteins. *Phys. Rev. Lett.*, 90(12):128102, 2003.

- [3] J. Elf D. Fange. Noise induced Min phenotypes in *E. coli*. *PLoS Comput. Biol.*, 2(6):e80, 2006.
- [4] M. Sturrock, A. Hellander, A. Marsavinos, and M. Chaplain. Spatial stochastic modeling of the Hes1 pathway: Intrinsic noise can explain heterogeneity in embryonic stem cell differentiation. *J. Roy. Soc. Interface*, 10(80):20120988, 2013.
- [5] M. Sturrock, A. Hellander, S. Aldakheel, L. Petzold, and M. Chaplain. The role of dimerisation and nuclear transport in the Hes1 gene regulatory network. *Bull. Math. Biol.*, 76(4):766–798, 2013.
- [6] K. Takahashi, S. Tănase-Nicola, and P. R. ten Wolde. Spatio-temporal correlations can drastically change the response of a MAPK pathway. *Proc. Natl. Acad. Sci. USA*, 107(6):2473–2478, 2010.
- [7] D. Fange, O. G. Berg, P. Sjöberg, and J. Elf. Stochastic reaction-diffusion kinetics in the microscopic limit. *Proc. Natl. Acad. Sci. USA*, 107(46):19820–19825, 2010.
- [8] S. Hellander and L. R. Petzold. Reaction rates for a generalized reaction-diffusion master equation. *Phys. Rev. E*, 93(1):013307, 2016.
- [9] J. Elf and M. Ehrenberg. Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases. *Syst. Biol.*, 1(2):230–236, 2004.
- [10] J. Hattne, D. Fange, and J. Elf. Stochastic reaction-diffusion simulation with MesoRD. *Bioinformatics*, 21(12):2923–2924, 2005.
- [11] R. F. Oliveira, A. Terrin, G. Di Benedetto, R. C. Cannon, W. Koh, M. Kim, M. Zaccolo, and K. T. Blackwell. The role of type 4 phosphodiesterases in generating microdomains of camp: Large scale stochastic simulations. *PLOS ONE*, 4(7):e11725, 2010.
- [12] M. Tomita, K. Hashimoto, K. Takahashi, T. Shimizu, Y. Matsuzaki, F. Miyoshi, K. Saito, S. Tanida, K. Yugi, J. Craig, V. Clyde, and A. Hutchison. E-Cell: Software environment for whole cell simulation. *Bioinformatics*, 15(1), 1999.
- [13] I. Hepburn, W. Chen, S. Wils, and E. De Schutter. STEPS: efficient simulation of stochastic reaction-diffusion models in realistic morphologies. *BMC Syst. Biol.*, 6(35):425–437, 2012.
- [14] S. S. Andrews and D. Bray. Stochastic simulation of chemical reactions with spatial resolution and single molecule detail. *Phys. Biol.*, 1(3):137–151, 2004.
- [15] R. A. Kerr, T. M. Bartol, B. Kaminsky, M. Dittrich, J.-C. Jack Chang, S. B. Baden, T. J. Sejnowski, and J. R. Stiles. Fast Monte Carlo simulation methods for biological reaction-diffusion systems in solution and on surfaces. *SIAM J. Sci. Comput.*, 30(6):3126–3149, 2008.
- [16] S. Engblom, L. Ferm, A. Hellander, and P. Lötstedt. Simulation of stochastic reaction-diffusion processes on unstructured meshes. *J. Sci. Comput.*, 31(3):1774–1797, 2009.
- [17] S. Hellander, A. Hellander, and L. R. Petzold. Reaction-diffusion master equation in the microscopic limit. *Phys. Rev. E*, 85(4):042901, 2012.
- [18] S. Hellander, A. Hellander, and L. R. Petzold. Reaction rates for mesoscopic reaction-diffusion kinetics. *Phys. Rev. E*, 91(2):023312, 2015.
- [19] R. Erban and J. Chapman. Stochastic modelling of reaction-diffusion processes: algorithms for bimolecular reactions. *Phys. Biol.*, 6(4):046001, 2009.
- [20] S. A. Isaacson. A convergent reaction-diffusion master equation. *J. Chem. Phys.*, 139(5):054101, 2013.
- [21] M. Doi. Second quantization representation for classical many-particle system. *J. Phys. A: Math. Gen.*, 9(9):1465–1477, 1976.
- [22] M. v. Smoluchowski. Versuch einer mathematischen Theorie der Koagulationskinetik kolloider Lösungen. *Z. phys. Chemie*, 92(2):129–168, 1917.
- [23] N. Agmon and A. Szabo. Theory of reversible diffusion-influenced reactions. *J. Chem. Phys.*, 92(9):5270, 1990.
- [24] A. Hellander, S. Hellander, and P. Lötstedt. Coupled mesoscopic and microscopic simulation of stochastic reaction-diffusion processes in mixed dimensions. *Multiscale Model. Simul.*, 10(2):585–611, 2012.
- [25] M. B. Flegg, S. Hellander, and R. Erban. Convergence of methods for coupling of microscopic and mesoscopic reaction-diffusion simulations. *J. Comput. Phys.*, 289:1–17, 2015.
- [26] M. Flegg, J. Chapman, and R. Erban. The two-regime method for optimizing stochastic reaction-diffusion simulations. *J. R. Soc. Interface*, 9(70):859–868, 2012.
- [27] M. Robinson, S. Andrews, and R. Erban. Multiscale reaction-diffusion simulations with Smoldyn. *Bioinformatics*, 31(14):2406–2408, 2015.
- [28] J. C. Jaeger H. S. Carslaw. *Conduction of Heat in Solids*. Oxford University Press, 1959.
- [29] J. S. van Zon and P. R. ten Wolde. Simulating biochemical networks at the particle level and in time and space: Green’s-function reaction dynamics. *Phys. Rev. Lett.*, 94(12):128103, 2005.
- [30] J. S. van Zon and P. R. ten Wolde. Green’s-function reaction dynamics: A particle-based approach for simulating biochemical networks in time and space. *J. Chem. Phys.*, 123(23):234910, 2005.
- [31] S. Hellander and P. Lötstedt. Flexible single molecule simulation of reaction-diffusion processes. *J. Comput. Phys.*, 230(10):3948–3965, 2011.
- [32] M. J. Morelli and P. R. ten Wolde. Reaction brownian dynamics and the effect of spatial fluctuations on the gain of a push-pull network. *J. Chem. Phys.*, 129(5):054112, 2008.
- [33] F. C. Collins and G. E. Kimball. Diffusion-controlled reaction rates. *J. Colloid Sci.*, 4(4):425–437, 1949.
- [34] D. T. Gillespie. A diffusional bimolecular propensity function. *J. Chem. Phys.*, 131(16):164109, 2009.
- [35] O. Tange. Gnu parallel: The command-line power tool. *login: The USENIX Magazine*, 36(1):42–47, Feb 2011.
- [36] C. Geuzaine and J.-F. Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities. *Int. J. Numer. Meth. Eng.*, 79(11):1309–1331, 2009.